1

USAISEC

US Army Information System Engineering Command
Fort Huachuca, AZ 85613-5300

U.S. ARMY INSTITUTE FOR RESEARCH
IN MANAGEMENT INFORMATION,
COMMUNICATIONS, AND COMPUTER SCIENCES
(AIRMICS)

AD-A218 300

System Considerations and Methodologies in Support of Software Development and Maintenance in the Information System Resource

(ASQBG-A-89-037)

October, 1988

AIRMICS
115 O'Keefe Bldg
Georgia Institute of Technology
Atlanta, GA 30332-0800

90 01 30 007

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704--188 Exp. Date: Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS NONE
2a. SECURITY CLASSIFICATION AUTHORITY N/A	3. DISTRIBUTION / AVAILABILITY OF REPORT N/A
2b. DECLASSIFICATION / DOUWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) ASQBG-A-89-037	5. MONITORING ORGANIZATION REPORT NUMBER(S) N/A
6a. NAME OF PERFORMING ORGANIZATION AIRMICS 6b. OFFICE SYMBOL (if applicable) ASQBG - A	7a. NAME OF MONITORING ORGANIZATION N/A
6c. ADDRESS (City, State, and ZIP Code) 115 O'Keefe Bldg., Georgia Institute of Technology Atlanta, GA 30332-0800	7b. ADDRESS (City, State, and Zip Code) N/A
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AIRMICS 8b. OFFICE SYMBOL (if applicable) ASQBG - A	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) 115 O'Keefe Bldg., Georgia Institute of Technology Atlanta, GA 30332-0800	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO.: 62783A PROJECT NO.: DY10 TASK NO.: 07 WORK UNIT ACCESSION NO.:

11. TITLE (Include Security Classification)

System Considerations and Methodologies in Support of Software Development and Maintenance in the Information System Resource (UNCLASSIFIED)

12. PERSONAL AUTHOR(S)

G.J. Dunleavy

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
	FROM ____ TO ____	1988 October	19

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Information System Resource, Army Information Architecture

19. ABSTRACT (Continue on reverse if necessary and identity by block number)

This report examines the system considerations and methodologies of software development and maintenance in the Information Systems Resource. Addressed are the shortfalls of the current Army Information Architecture, the relationships between hardware, software and transport facilities within the current information architecture, the use of structured and rapid prototypes, and the relationship of software development and technology insertion.

(KP)

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED / UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Michael J. Evans	22b. TELEPHONE (Include Area Code) (404) 894-3107 22c OFFICE SYMBOL ASQBG - A

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted

All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

This research was sponsored by the Army Institute for Research in Management Information, Communications, and Computer Sciences (AIRMICS), the RDTE organization of the U.S. Army Information Systems Engineering Command (USAISEC). This report examines the system considerations and methodologies of software development and maintenance in the Information System Resource. This research report is not to be construed as an official Army position, unless so designated by other authorized documents. Material included herein is approved for public release, distribution unlimited. Not protected by copyright laws.

THIS REPORT HAS BEEN REVIEWED AND IS APPROVED

s/ James Gantt
James Gantt, Chief
Management Information
Systems Division

s/ John R. Mitchell
John R. Mitchell
Director
AIRMICS

Table of Contents

1. INTRODUCTION

This task deliverable, Task #1, is targeted at articulating the system considerations and methodologies of software development and maintenance in the Army Information Systems Resource. Prime requisites for both development and maintenance are the Army goals of Mission, Modernization, and Standardization.

Traditional developments have attempted to trade-off the seemingly diametric goals of modernization and standardization. As the specific goals of modernization are mapped to requirements of the user mission, standardization is often sacrificed. As the specificity and complexity of the mission requirement grows, the ability to maintain standardization wanes. Theory states that, " After all, the mission must be accomplished, and, then, standards can be addressed. "

Constraints of available development languages, software tools, non-standard data, an evolving architecture, and systems in varying stages of production, deployment, and planning make the ability to map to standards less likely. Moreover, Peace-Time and Go-To-War missions represent very vastly different environments.

Given that the mission must be performed, adherence to standards to support integration and modernization is best served by severing the trade-off connection. If standardization is not viewed as a diametric to mission requirements, adherence becomes a function of how do we do it, rather than we can't give up the mission to achieve standardization. Mission change is a <u>given</u>. There is no magic, single product which will <u>ever</u> successfully interface directly with existing 2GL, 3GL, 4GL, and nGL systems to come.

Once the diametric connection is broken, the need to use tools and elements embedded directly in the mission software suite vanishes. The key to addressing standards, while achieving mission requirements, is to use available data, and to vary the interface via technology insertion of <u>"bridge"</u> products. <u>"Bridge"</u> is defined here as an interface element used to gain access to data. It is not intended to describe the more specific, popular use of the term in a communications environment which is that of a Data Link Layer connection, as opposed to routers and/or gateways. Bridge products can establish interfaces at the level of Mission, be it Base Sustaining, Tactical, or Strategic. Missions may be viewed as user requirement driven systems dedicated to producing data for analysis, action, manipulation, and

reporting. The bridge products may be viewed as <u>available data driven</u> systems dedicated to facilitating standardization.

When the bridge has no mission-dependent function, it may serve as a decision analysis tool. As such, it becomes the base for development of prototypes which can drive models to simulate conditions and outcomes. It's domain is not governed by mission-stated requirements. Instead, it can function as a base element in a Decision Support System (DSS) that can support and enhance mission objectives.

Modern computer environments affirm the need for DSS. Automatic Data Processing (ADP) development cycles frustrate users. Although modern processors and software tools provide unprecedented gains in instruction execution, data storage, transaction processing, and report production capabilities, users may be less satisfied than ever.

Users protest that the ADP staff doesn't listen to requirements; ADP personnel argue that users don't know what they want. James Martin in his " Information Systems Manifesto " describes a nameless decision maker as follows, " I don't know what I want, but I'll recognize it when I see it. "

In fact, user's wants change from request to request. In one instance, summary data may satisfy his information need. In other instances, he may need to make a complex analysis of multiple data elements that requires a relational database. He may need to use one result to generate another information request. His typical " what-if " requests require a flexible system with a rich tool set that provides him with different views of data. Data extraction capability for use in a DSS is a good realization of these diverse information demands.

Another important aspect of user information requests that adds to ADP provider-user frustration is that of timeliness. Users at all levels have periodic needs to gain immediate access to data for analysis, planning, and day-to-day operations. While the data elements and data types are available, there is no quick-response method to meet the user request. Again, a DSS with a tool set that has access to extracted data could readily satisfy these requests.

Moreover, bridge product data extraction offers significant opportunities for the development of Audit Trails and has high potential for component reuse. The most important aspect of the bridge is that it perform its function in a near transparent mode. Although it serves no immediate mission related function, it also can <u>not</u> levy heavy burdens on mission resources.

2. <u>OVERVIEW</u>

2.1. Problem Definition

The goals of the Army Modernization and Mission Support are rooted in doctrine and are the capstones for development. To move towards these goals, it is necessary to develop a plan that:

- Supports missions
- Utilizes existing resources and assets
- Provides flexibility to react to change and emerging requirements
- Permits and encourages standard development practices
- Establishes and demonstrates milestone achievements
- Creates and maintains reusable components
- Achieves data sharing
- Delivers on the "More Bang for the Buck" opportunities

The present mix of fielded systems, systems under development, and planned systems offer a number of challenges in the search for commonality required by the plan objectives stated above. Systems run the gamut from 2GL batch-oriented, procedural language, flat-file, stand alone systems to the latest 4GL on-line, real-time systems written in object-oriented languages (ADA) that use relational databases. To analyze these systems, the view <u>must</u> be taken from above the mission.

From this view, commonality becomes more distinct. Each of the missions or combination of missions performed in a given system consists of a number of processes that utilize data and/or transaction input to conduct the mission. Each mission is also defined in an Information Management Plan (IMP). IMPs describe mission elements and processes. Whether the IMP resides on paper, in a database, or a combination of both, a mission description exists. Analysis at the system level is the key to objective examination, based on commonality.

Each system is composed of processes and executes in a given configuration of hardware, software, and transport. The hardware, software, and transport components have certain characteristics and requirements. Missions demand specific performance. Given a stated configuration, specific processing element descriptions, and the performance criteria, <u>all</u> systems have commonality.

Since it is also a given that Army systems must migrate

towards a current ideal of on-line, relational database systems written in ADA, the bounding constraints of current and target goals are established. The plan must develop and implement the technology necessary to migrate from the current to the target environment.

Given the base and desired goals, primary tasks necessary to analysis are:

- Capture of configuration data for:
  - Current systems
  - In-progress upgrades
  - Future targeted systems
- A set of "Smart Tools" which permit system managers, system engineers, analysts, and developers to make intelligent decisions about the migration.
- Sound practices for establishing and maintaining audit trails.
- Proof-of-process that development tool products produce the expected results.
- Development and implementation of a Migration Plan.

The conceptual detachment from mission operations depends upon a useful, non-disruptive bridge interface. The bridge must address and solve a number of interrelated problems. Among the more significant problems are:

- Impact on the three tier architecture
- Impact on life cycles
- The building and maintenance of Mission-Independent Data (Meta-Data)

2.2. Impact on Three Tier Architecture

The HQDA Information Model, developed in accordance with AR25-1 and AR25-5, is the framework that defines all Information Mission Area (IMA) relationships of Information Management. The Architectural Model is a top-down logical structure that serves as the host for all subservient information architectures to standardize Data, Application, and Geographic/Technical Architectures.

Bridge product technology insertion specifically meets the stated requirements for Standard Tool Development, Data Standardization, Cost Reduction via Reuse, and migration to Standard Application Development.

By its nature, technology insertion supports Regional Service Centers, Organizations, and Users, in accordance with AR 25-1 and the Three Tier Architecture. Technology insertion also ensures that the user needs of system access, data access, and decision support are met.

Insertion facilitates both functional and technical integration efforts.

2.3. Impact on Life Cycles

Successful bridge development extends product life cycles. Its Follow-On-Processor design ensures that it utilizes mission-developed data, rather than bending mission requirements to get data. A critical design consideration is the assurance that all applications offer interface boundaries. Regardless of implementation language constraints and application structure, software facilities like bridges can utilize effectively these interface points.

An apt analogy is that of the use of the personal computer. Because mainframe users once viewed PCs as toys that wouldn't deliver significant results and that also posed no threat to their kingdoms, departmental users were cleared to acquire them. When they became ubiquitous, users found ways to extend their capabilities. Modems, terminal adaptors, workstations, LANs, and hardware/software interfaces to circuit, packet, ISDN, and message switches all use bridge philosophy to extend PC use and life cycles. Widespread use of X.400 will again revolutionize the use of the PC. Indeed, the engine in the PC itself is being redesigned to bridge to mainframes and super-computers.

Prototyping using 4GL, screen generators and DBMS systems offers opportunities to extend life cycles and to reduce costs. Furthermore, AR 25-5 recommends that prototypes be used for problem definition, evaluation, testing, and verification and validation of proposed solutions.

Prototype development can dramatically decrease, if not obviate, the time necessary to develop and approve preliminary documents in the early design stages. The most compelling reason for prototypes, however, is to ensure the correctness and validity of results gained by using modern DBMS and 4GL tools to discover bugs early. E. F. Codd, President of the Relational Institute of San Jose, and considered by many to be the "Father of Relational Technology", has published a special, two part report in the August and September 1988 issues of DATAMATION entitled "Fatal Flaws in SQL", that speaks to this need.

2.4. Toward Mission-Independent Data (Meta-Data)

While data is always represented in a structure and format that best serves the application goals, it is not the only possible view of the data. Advances in DBMSs, especially in the area of Structured Query Languages (SQL), demonstrate this point. Once data is successfully extracted via a bridge element, the view of the data is no longer constrained by its application values. DSSs, using Expert or Artificial Intelligence views, may take a totally different view of the abstract data. All data is an abstraction and the view of the abstraction differs from where and for what purpose it is viewed.

The salient point is that if the view is not mission dependent, the data offers different opportunities for use.

3. SHORTFALLS IN THE CURRENT ARMY INFORMATION ARCHITECTURE

AR-25-1 Provides goals and objectives for an Army-Wide Architecture and a support structure for Information Management.

3.1. Shortfalls

The most critical shortfall in the Army Information Architecture and the companion Data, Application and Geographic/Technical Architectures is the notional sense of the program and the naivety implied in reaching the objective and target stages of the program.

The second shortfall is the failure to establish starting and ending points of reference, particularly in respect to reaching a capability to process and manage data at a Meta level.

The third shortfall is the concept of a Three-tier Architecture. Three-tier architecture has some relationship to the present configuration. The relationship is a direct result of how the Information Systems Resource evolved in the Army and the fact that the vast majority of current assets are employed in a fashion consistent with the vintage of the hardware, software and transport facilities available at the time of acquisition.

3.2. Discussion

The technical characteristics of the processing assets

throughout the Army are tied to the 2nd, 3rd, 4th, and 5th generation processors. The vast majority of high-end assets are best characterized as 3rd generation.

The dominant characteristics of the current inventory is either stove-pipe, stand alone or Batch/Batch Inquiry. The majority of records are stored as flat files. The systems are normally responsive to user needs in Peace-Time, yet one should seriously question the ability of the resource to successfully support a full scale mobilization and a large scale conflict.

One might choose to point out that the configuration which is on the ground today supported a large scale effort in the Republic of Vietnam (RVN). This is quite true, but the situation in RVN did not require the Combat Arms to be as maneuverable as they would need to be in an engagement against the Warsaw Pact. The opposition in RVN did not posses the ability to disrupt or destroy all levels of resources. The enemy in RVN did not target Information Systems Resources particularly those which could be used as part of a Communications Infrastructure at a later date. The rate of expenditure of resources nor the demands for immediate intelligence simply were not there.

The battle doctrine has changed, but the Information Systems Resource has not changed at the same pace and considerable improvements are dictated if we are to be able to deter a numerically superior modern fighting force who also has considerable capability in the electronic warfare arena.

A support plan for the migration of critical mobilization and war fighting information systems assets must be identified and prioritized at the top of the Planning, Programing, Budgeting, and Execution System (PPBES). The list will include systems in the theater/tactical, strategic and sustaining base. System as used in this case means hardware, software, transport and the SOP, documentation, and training needed to support and maintain the effective combat readiness.

The resultant plan should not produce a new set of stove-pipes or multiple layers of unique software. Performance should precede functionality on a rating scale for these systems. Timeliness and accuracy of the information and the devoted products must be at the forefront of design decisions. The fact that a system performs the function will not suffice. The government should hold the providers feet to the fire for the performance and quality of the hardware, software, and communications.

Some differentiation must be made between systems which require high performance and others where routine services will suffice. The architecture process should accommodate and

account for the nature of the data transiting the system.

CONUS is the natural base of supply operations for assets of the Army. The OCONUS Theaters are extensions of the CONUS base regardless of the command authority given to the Theater Commander. It is impossible to give an OCONUS command infinite resources.

The resource resupply base is, and must be, a centrally managed asset. The Central Manager must allocate resources to geographically dispersed locations. The Central Manager requires a running view of status at all levels of the supply chain. Additionally, the Central Commodity Managers need information about the physical movement of the commodity being managed. In the ultimate, a system which provides a Just-in Time (JIT) operation, satisfies the user needs and eliminates many burdensome costs and logistics associated with stock piles. Information and Data Management should be just like any other commodity.

The Architectural process should not begin with a preconceived notion that "n" tiers are appropriate. The model may begin as a three-tiered organization, but in order to satisfy needs, we require a great deal more information than what is connoted by a three-tier process devoid of performance characteristics.

3.3. Strength

The goal of Data Standardization, Synchronization, and the objective of reaching a processing and management of Meta Data is excellent and is the most important strength in the Army Information Architecture.

3.4. Discussion

The questions are: How you accomplish this? How long will it take? How much will it cost?

After several years of work on: Army Corporate Data Base, Data Dictionary & Encyclopedia, Army Information Engineering, Reserve Component Automation System, STAMIS Modernization, the question of how many data elements does the Army use goes unanswered. The same is true for what the configuration of the current system, in progress and planned systems.

The targets are still moving. The Army is applying resources pell-mell without the benefit of a prioritized action plan or so much as a detailed view of the current resource configuration.

AR25-1's underling premise that Data Standardization provides

the key to modernization and improvement has probably skewed the logical processes out of order and out of proportion.

ACDB , AIE, AD&E are all drawing more resources than is ARPMIS. They still do not work but ARPMIS does, albeit with less than optimum performance levels.

Several IMP, IMMP cycles have gone by since the issuance of the order for the first IMMP study. How much of the data from the IMMP has been ported to the ARPMIS Data Base? DSI has been unable to obtain an answer to that question. If the answer is none or very little then the upper echelons of the Information Systems management community should be asking some very pointed questions reference the intent and purpose of the IMMP. Clearly, the IMP/IMMP assists the staff levels in preparing the PPBES. Is that all there is? How is the information content used for other than the budget exercise? What benefits have accrued for the MACOMs? What is the plus/minus picture at the DOIM level? The exercise is man-power intensive given that there are no automation tools assigned to the task. Does the DOIM get back planning information to aid him in managing the Information Processes at his level? If not, DSI suggests this is a serious flaw in the process.

The success or failure of a program of the magnitude alluded to in AR25-1 requires cooperation and coordination at all levels if it is to succeed. The program goals are laid out in AR25-1 and it is reasonable to expect that the MACOMs and DOIMs have some expectations as a result of the publishing of AR25-1. There exists a credibility issue between the DA Staff and the action officer levels. The program goals allude to some very specific improvements in the form and function of information delivery. Specific references to key programs leading to the improvements were made. CAMIS/RCAS, STARNET, ACDB, STAMIS Modernization, et al. Where have these programs gone since the issue of AR25-1? Have any of these programs provided improvements in the real-world operating environments or are they just another set of Trojan Horses?

3.5. Conclusion

A conclusion can be drawn at this point. It is that the modernization program lacks credibility. The reasons for the credibility gap can be traced to one of these problems:

- A poor or incomplete concept plan
- Lack of follow-up action and audits on the impacts of actions
- Total decay of resource support or misdirection and application of resources.

DSI believes the problems arise as a result of dictatorial

application of unrealistic goals without benefit of technical guidance at to how one is to reach these goals. The most difficult elements of the modernization objectives and goals are their fuzziness. The fuzziness is understandable up to the point where a study captures the configuration data on the system as it stands on-the-ground, and does like wise for in process and planned changes.

The process should then be one of analysis, modeling, simulation to determine the best course of action and to establish real priorities, time-lines and resource requirements. The processes are not organized nor are they controlled.

The first priority should and must be the gathering of "all" of the configuration data relating to the present system. Gathering meaning capture of the information in an electronic form so the data can be analyzed and a model constructed. Once the model or models are constructed, technology tools in the form of DSS should be employed to determine the level of performance of the present system and to identify shortfalls in both functionality and performance. AI based decision support tools should be used to perform trade-off and what-if types of exercises.

The actual model or sets of models will not necessarily resemble a three-tiered operation. Many of the processes will show up as stubbly pencil and tennis shoe interfaces. This is not a negative. It is a true reflection of how the real-world system works. A cross section view of the resultant model will dissolve the presence of assets assigned to one or another process which could be shared to satisfy shortfalls in other areas.

The analysts, engineers, and programmers need a road map, not just a concept. The road map must be based on the real world situation. In the real world, some processes serving some users are more important, more critical, and more time sensitive than others. Some processes are very well suited to a batch processing environment, now and forever. Those processes which are best done in batch should be identified. Those which are not should be separated from the batch processes.

The question at this point is "Who is going to provide the road map?". The DA Staff are in a position to influence the direction to be taken by the road map but not the detailed information about how to negotiate the terrain features found on the ground. The user may appear to be the key figure since he knows the detailed nitty-gritty of the requirement. After all it is his requirement. To allow the user to draw the road map would cause major problems. To borrow and modify an old cliche, "The user cannot see the forest for the trees.". He is too close to the problem. Who then should

determine the real world situation and who should draw the road map? The System Manager is the only player in the right position to be objective enough to determine these answers. Before he draws that map however, the Systems Manager must take a long hard look at himself and his performance record. The most important step he can take now is to commit himself to the more efficient use of his resources. This is done by demonstrating a real commitment to applying innovative technology where is makes economic sense. Just as bad news does not get any better with age, the problems in the ISR are neither going to go away nor are they going to get better with age. If the Systems Manager continues to not take action on this problems, the users and the DA Staff will be forced to continue trying to draw their own road maps and use them in their own "real world".

After the analysts, engineers, and programmers get their road map what other considerations deserve their attention? Contention and overhead are two factors which require a great deal more attention than they presently get. As more and more STAMIS applications transition from the batch to real-time, the demand for central processing unit cycle time goes up. As the real-time or near real-time functions siphon off cycle-times, the efficiency losses accrue to the batch-operations.

Tuning and Optimization must take place at each of the cooperating processing nodes. At any given point in time, the ISR has a finite capacity to process work. The value of this capacity is some number "X". The number moves from its maximum optimized condition downward as the number and mix of tasks vying for service change. No real number value exists for "X" nor do we have the resources in hand to identify the boundaries of these numbers.

This author has often heard the question asked "What processing power is being brought to bear on the overall processing needs by the more than 100,000 PC type devices which have entered the inventory over the past five years or so?". The answer is probably very little. DSI is not aware of any definitive study which would lead to a quantifiable answer. Several years ago, this author reviewed a Masters Thesis on Office Automation in the Army. The author stated that 2% or less of the PCs in the office were used and the bulk of the capacity was used in word-processing.

This author has also seen references to processing capacity made by auditors and congressional staffers which allude to the fact that the Army is over subscribed with processing power and simply has not properly managed available assets. The foundation of this line of logic is tied to the comparisons drawn between 1960 generation 360 class CPU's and the cycle capacity of CPU's employed in PC level devices. If that were in fact a good analogy, the Army might be over

subscribed on processing capacity. The truth is, cycle speed is not a good basis to establish a 1 to 1 comparison. A single user PC can not bring to bear the full capacity of it's processing power for extended periods of time. The CPU spends far more time waiting than it does working. The total machine architecture, software, and I/O capacities are as different as the abacus was from a UNIVAC I or a 360.

This commentary is not intended to impugn the technical acumen of the auditors, nor the congressional staffers, but in fact, they published and circulated a report with this type of data which has the effect of adding bias to the non technical decision makers. The Army must provide the non-technical decision makers with a view of how much processing power is enough, and in what form and where and how it will be managed. Capacity planning is the key.

The Army is trying to eat the problem as though it were one problem, and not a list of problems. The Army Information Architecture is flawed from start to where ever. (There is no logical conclusion) The notion that the Army can migrate from the present system without a detailed analysis of the present system is unrealistic.

The dominant reality is there is not enough money available, nor is the DA STAFF the proper place at which technical management decisions can be made.

The DA STAFF has signed up to Data Management on too grand of a scale, and without proper consideration of the support environments of hardware, software and transport which must proceed the management of Meta-Data.

The DA STAFF should set the goals and establish target time-lines. Detailed analysis of the goals, time frames and resources required to meet the stated goals should come from the technical managers.

A logical first step in this process might be the modernization and upgrade of resources tasked with managing the ISR.

4. <u>RELATIONSHIP BETWEEN HARDWARE, SOFTWARE, AND TRANSPORT FACILITIES OF THE CURRENT ARMY INFORMATION ARCHITECTURE.</u>

What is meant by the relationship between hardware, software, and transport facilities in the current Army Information Architecture? Most tend to think of these parts of an information system as separate entities. We buy hardware from one source, we buy application software from possibly another and we go to ISC/ISEC for our communications

requirements. We plan and purchase the components of information systems piece-meal, but we expect it to operate as a system. If it ever does operate as a system, it is very unlikely that it will ever perform up to either its own potential or our expectations.

The players in this scenario are the STAMIS representing top-down driven requirements and the command uniques or Installation Support Modules (ISM) and small users representing the bottom-up driven requirements. It is obvious that we are all marching to the beat of a different drummer, somewhat by necessity, but it is also obvious that most of us are also out of step with each other. Traditionally, we have "thrown hardware at the problem" seeking easy solutions. We have purchased literally tons of off-the-shelf and custom developed, but difficult to modify software. We are forever demanding more communications. Yet we are almost as far from a system solution as we ever were. Why? What are we going to do about it? How are we going to it? When are we going to do it?

As we address these questions we will begin to see the need for an increased sensitivity to optimizing performance. Optimization must involve three components of Information Systems; hardware, software, and transport.

4.1. System/Subsystem Level

Looking at the problem from the system/subsystem level it is easy to identify situations where parts of the Information System do not work together well or do not perform as well as should be expected. The STAMIS run on hardware that has tremendous processing capabilities at the ASIMS level, but since we essentially run forty plus (the number of installations supported by ASIMS) separate and different data bases, mostly in a batch/batch inquiry oriented mode, we do not get optimum results. As ASIMS is brought onto DDN, the transport component of the STAMIS will be improved between the Regional Data Centers (RDC) and the installations. STAMIS which have been ported over to PCs borders on the ridiculous when viewed from an optimum performance perspective.

The ISM are the most optimized systems of the three players primarily because they are newer than STAMIS/ASIMS. Essentially they are closer to the problem and have been able to more efficiently optimize their hardware and software. As currently configured, they have a relatively small transport function requirement because they are physically located on the installation they support.

Users present a different kind of problem than either the STAMIS or the ISM because they are both more numerous and

less organized. Users tend to develop solutions in a vacuum. They seldom ask for or consider input from others. They frequently use whatever hardware and software is available, even is it is wrong from a systems point of view. If the need exists to communicate with external systems, users tend to act as if theirs is the only request for communications that ISC is processing. They have tunnel vision as it were because they do not see the total picture of many users collectively asking for more communications capability than exists. ISC lacks the tools to effectively deal with this problem and it also lacks an effective method of communicating information back to the user about how ISC will provide the solution to the requirements. One of the reasons for the long lead time in circuit acquisition process is that the current methods used by ISC and DCA require a lot of manual administrative time to determine how to satisfy the users needs as economically as possible.

Relationships leads to interdependences. All three components of Information Systems must work together for the user to really benefit. Almost all of the time users systems perform at suboptimal levels. They simply do not know how to obtain an optimum solution.

4.2. Performance Optimization

Why does one need for these relationships to be optimized? What happens if they aren't? What is the impact? Who cares and why?

Generally, STAMIS personnel will be sensitive to optimizing hardware and software for their particular applications program. Some increase in performance can be achieved with relatively little cost such as running the CPU at a higher clock speed but significant improvements are more difficult to reach. The applications software which has been developed for the STAMIS is difficult to manage and to modify. It has been developed in older languages and sometimes in unstructured methodologies. STAMIS Modernization is the step necessary to realize significant increases in performance. However, as was previously discussed, STAMIS Modernization will not be a reality any time soon.

This is not as true for user uniques. Their problems are relatively small in comparison and they are using more modern tools to satisfy their needs. What do these people do to optimize performance with current versions? Today there is very little that users can actually do if they do not solicit aid from ISC/ISEC. The command uniques have shown the most promise of optimization at this time because they do not have as far to go developmentally as either the STAMIS or the users do.

If the DOIM does not analyze the users needs vs. the communications impact, who will do it from a communications point of view? More importantly, who does from a systems point of view? The system manager is the responsible agent for the review. His primary concerns are in the communications area and he does not always get involved with hardware and software relationships, performance and acquisition. This leads to a tacit acceptance of systems with less than optimum performance designs. In the long run this will cost more money, more communications money.

4.3. System Manager

The only way for the system manager to fairly (according to mission needs) apportion those resources he controls is for him to continuously assess what the STAMIS , the command uniques, and the users are doing. Before they are allowed to use the system they must have and use certain common elements such as standardized communications protocols and data elements. Examples are the requirement to use ADA in all new and major redesign systems and the requirement to use GOSIP and POSIX.

Once approved data elements need to be protected. The Army must control input to STAMIS

Security of these large and growing systems is a critical issue. Systems High and Controlled Access are useable techniques but until true multilevel security is a reality we can only come close to optimum performance.

Because of the continued growth of computing resources and the cap on communications usage, the competition for the transport resource will heat up. We must take steps to optimize the use of our fixed assets. ARPMIS is the key tool available to the systems manager to aid in doing the analysis and modeling necessary.

There are across the board problems in the strategic, tactical, and sustaining base areas. The form and function of STAMIS is not uniform or constant across all three. They use different hardware and different transport. Functionally they are almost the same but performance is no where near the same. TACCS is extremely limited in its performance while running a STAMIS. The TACCS peripherals are improving but are not yet sufficiently mature.

The forms of STAMIS are different. Given that difference, there remains the need to optimize all STAMIS for performance. Taking the step of optimization will force us to improve the STAMIS uniformly.

5. PROTOTYPING METHODOLOGIES

Two principal forms of prototyping methodologies enjoying widespread use are Structured and Rapid.

5.1. Structured

In the structured form of prototyping, a top-level methodology hosts the prototyping technique to aid in systems development. In the instant case, prototyping is hosted by the methodologies directed by AR 25-1, AR 25-12, DOD-STD-2167A and DOD-STD-7935(.1) for ADP systems development. The prototype is used as a tool throughout the phases of the development life cycle.

As noted above, prototyping aids in the early identification of system problems. Another important benefit of prototyping is that it takes the "requirement" abstraction and reduces it to a visible, if not tangible, example of how the system will work. Users and developers have scaled replicas of the system that use representative data to demonstrate system functionality and performance. This pilot operation offers the best opportunity for resolving user-ADP developer communication difficulties.

The prototype may become the base for an enhanced prototype or may be used in similar prototype operations in other phases of development or other developments. The key is that the prototype is integrated as a development tool within the framework of the established methodology.

5.2. Rapid

In Rapid prototyping, the key is to build an initial prototype quickly to demonstrate some subset of required capability. Once the prototype is built and executes, it becomes the base for either an Evolutionary or Throwaway style of development. In evolutionary, prototype results become the base for enhancing the prototype to evolve towards stated requirements. This approach gains quick results, <u>but</u> it also offers several high-risk potentials. The first is that development is <u>not</u> subservient to a guiding, proven methodology. Development often turns to the improvement of the prototype processing and loses sight of the initial system goals of meeting requirements. Without a structured methodology, it's often difficult to identify and turn off the requirements definition phase.

A second form of Rapid prototyping is that of Throwaway. Using throwaway, a base premise is used to build the prototype and demonstrate the feasibility of some limited subset of requirements. When the feasibility is established, the prototype is not used as a base for development of further prototypes or system components. Throwaway prototyping can demonstrate requirements deficiencies or omissions but it always represents incremental costs that must be offset by the successor methodology. An excellent use of throwaway prototyping was when PC configurations were not able to support prototype memory, data, and execution requirements and the user had no access to a mini or mainframe.

5.3. Audit Methodologies

One of the best audit methodologies available is the use of DSS at the Software Development Centers (SDC) in development work. One of many possible measures of performance would be the number of good lines of code per day and the efficiency of that code.

5.4. Verification and Validation

As alluded to throughout this white paper, early verification and validation of requirements is the cardinal benefit derived from prototyping methodologies. Failure of traditional development cycle techniques to provide early V&V spurred prototype development.

5.5. Reuseability

As software development and maintenance costs become an increasing percentage of ADP budgets, component reuse is imperative. With the advent of non-procedural code and the ability to view data via structured queries, prototyping; modeling; and simulation elements are prime candidates for developing reusable components.

5.6. Development Tools

The data extraction, prototyping, and modeling elements used in the development of a migration plan to aid modernization should be catalogued into a Standard Development Tool Library for reuse.

5.7. Bridge Products and Methodologies

The base premise of this White Paper is to use bridge

products and smart tools to build an interface to achieve modernization and standardization. This ensures that the data extraction is non-disruptive to Mission performance and that it conforms to the Army Top-level Methodologies.

6. RELATIONSHIP OF SOFTWARE DEVELOPMENT AND TECHNOLOGY INSERTION

6.1. 2GL Bridge (Stand-Alone)

Second generation, fielded systems interface will require data extraction for Follow-On-Processing.

6.2. 3GL Bridge (Embedded)

For 3GL systems, bridge tasks written in the procedural language may be FORKed, SPAWNed, or otherwise created to accomplish the data extraction.

6.3. 4GL Generic Component

6.4. 5GL Library Facility

4GL & 5GL systems supporting non-procedural languages and Database query capabilities allow the use of "smart" database activity. "Smart" is defined as the ability to support concurrency and to provide processing Agents with the ability to use an Automated System Navigation Scheme. Characteristics of these to-be-developed tools would support inclusion as a System Generic Component or provide access via a cataloged Library Facility.

7. CONCLUSIONS

Mission performance remains the only job in the Army. Modernization, standardization, and development are subservient to conducting business.

ISC/ISEC can not avoid satisfying the mission requirement, but they only have finite resources.

Every Army application, fielded, in process, or on the drawing board can benefit from a successful Decision Support System implementation. Existing system descriptions form the basis for initial requirements definition. Systems under

development are subjects for study to plan effective technology insertion opportunities to support the mission. Planned systems offer pilot project prototype candidates for proof of process operations.

All systems must have documented descriptions

All systems offer interface points.

The goals of modernization and standardization are documented.

Technology insertion via bridge products can be made non-disruptive to mission performance.

4GL non-procedural code and advanced database capabilities are available.

Prototyping, modeling and simulation offer significant opportunities to extend life cycles, reduce costs, and produce flexible, extendable systems.

The Systems Manager must be the one to solve the problem of evolving from a hardware, software, and transport configuration which is limited in its flexibility and responsiveness (the present systems) to one in which these system elements are all optimized one to the other (the target or planned systems).

8. RECOMMENDATIONS

Establish AIRMICS as the ARPMIS Center for the development of a Decision Support System (DSS) for inclusion in the "All-Source Data Base described in Task #2.

Develop the Migration Plan to achieve modernization and standardization proposed in this White Paper.

Use Structured prototyping to demonstrate ability to meet requirements and to achieve Verification and Validation.

Select pilot project(s) for the proof-of-process of the data extraction and DSS capabilities.